International Center for Free and Open Source Software

Arduino-Based Humidity and Temperature Monitoring with SHT20 Sensor

Azwa Harshad

Internship

Open IoT

MARCH 2023

Contents

List of Figures **ii**

1 Introduction **1**

1.1 General Background 1

1.2 Working Principle 2

1.3 Components Needed 2

2 Circuit Diagram **4**

2.1 Circuit Connections 4

2.2 Block Diagram 5

3 Implementation **6**

3.1 Procedure 6

4 Code **7**

4.1 Code integrating SHT-20 sensor with arduino 7

4.2 Serial Monitor Output 9

5 Result **10**

5.1 Result 10

List of Figures

1.1 SHT20 sensor 1

2.1 Circuit Diagram 4
2.2 Block Diagram 5

4.1 Serial Monitotr 9

Chapter 1

Introduction

1.1 General Background

This focuses on humidity and temperature monitoring utilizing the Arduino platform alongside the SHT20 sensor. This project aims to provide an accessible and precise solution for tracking environmental conditions in diverse settings, from home automation to industrial applications. By integrating the SHT20 sensor with Arduino, our system offers real-time data on humidity and temperature, empowering users to make informed decisions and develop efficient control systems.

Figure 1.1: SHT20 sensor

1.2 Working Principle

The working principle of this project revolves around the interaction between the SHT20 sensor and the Arduino microcontroller. The SHT20 sensor utilizes a capacitive humidity sensor and a thermistor to measure humidity and temperature, respectively. These sensors detect changes in capacitance and resistance, which are then converted into digital signals. The Arduino microcontroller reads these digital signals via the I2C communication protocol, interpreting them as humidity and temperature values. The Arduino processes this data and presents it to the user through a connected display device or the serial monitor. By continuously sampling and processing data, the system provides real-time monitoring of environmental conditions, enabling users to track changes and make informed decisions based on the measured values.

1.3 Components Needed

1. Arduino Board (e.g., Arduino Uno)
2. SHT20 Sensor Module
    - Measurement Range:

      Humidity: 0percentage to 100percentage RH (Relative Humidity)

      Temperature: -40°C to 125°C
    - Accuracy:

      Humidity: ±3

      Temperature: ±0.5°C (typical)
    - Resolution:

      Humidity: 0.01

      Temperature: 0.01°C
    - Supply Voltage:

      2.1V to 3.6V (typically powered by 3.3V)
    - Communication Interface:

I2C (Inter-Integrated Circuit) digital interface

- Operating Current:

  Typical: 350 μA

  Standby: 0.5 μA

- Response Time:

  Humidity: 8 seconds (tau63

  Temperature: 5 seconds (tau63

- Size:

  Typical dimensions: 3 mm x 3 mm x 1.1 mm (LxWxH)

- Interface Compatibility:

  Fully compatible with Arduino and other microcontroller platforms supporting I2C communication.

- Calibration:

  Factory calibrated

- Long-Term Stability:

  Excellent long-term stability

3. LED

4. Breadboard

5. Jumper Wires

Chapter 2

Circuit Diagram

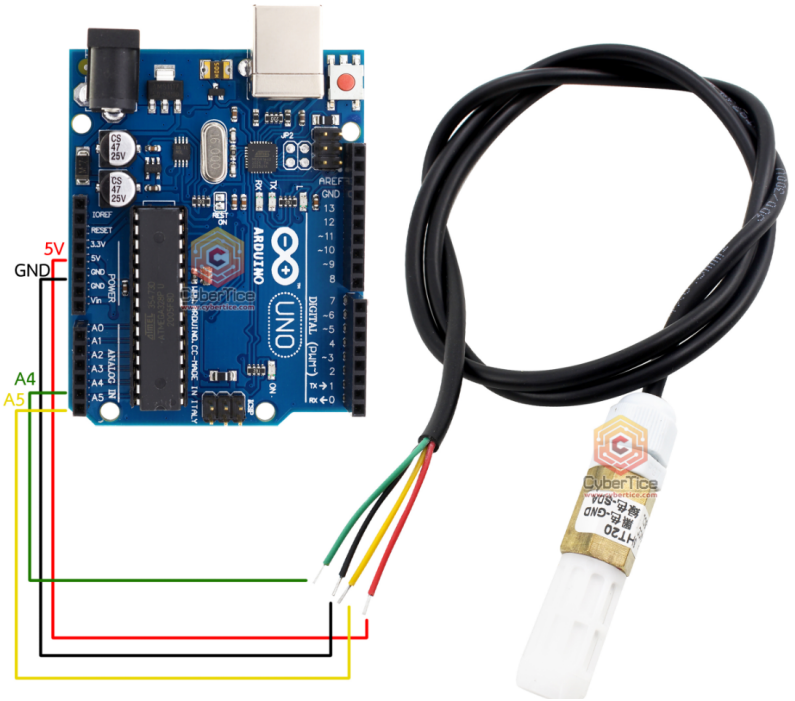

Figure 2.1: Circuit Diagram

2.1 Circuit Connections

1. Connect the VCC pin of the SHT20 sensor to the 5V pin on the Arduino.
2. Connect the GND pin of the SHT20 sensor to any GND pin on the Arduino.
3. Connect the SDA pin of the SHT20 sensor to the A4 pin on the Arduino.

4. Connect the SCL pin of the SHT20 sensor to the A5 pin on the Arduino

2.2 Block Diagram

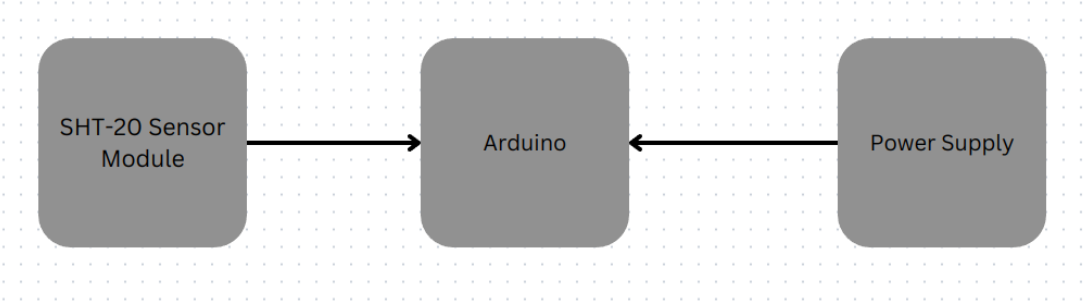

Figure 2.2: Block Diagram

Chapter 3

Implementation

3.1 Procedure

1. Setup Hardware: Gather all necessary components. Connect them according to the circuit diagram.

2. Install Arduino IDE: Download and install Arduino IDE from the official website.

3. Open Arduino IDE: Launch the Arduino IDE software.

4. Write Code: Compose your program using Arduino programming language (based on C/C++).
   Write setup and loop functions.

5. Verify Code: Click on the Verify button (checkmark icon) to check for any errors in the code.

6. Upload Code: Connect Arduino board to the computer via USB. Select the correct board and port from Tools menu.
   Click on the Upload button (right arrow icon) to upload the code to the Arduino board.

7. Test: Make sure the hardware is powered on.

Chapter 4

Code

4.1 Code integrating SHT-20 sensor with arduino

It is done through ArduinoIDE

Code

```
#include "DFRobot_SHT20.h"


/**
 * Hardware Connections:
 * -VCC = 3.3V
 * -GND = GND
 * -SDA = A4 (use inline 330 ohm resistor if your board is 5V)
 * -SCL = A5 (use inline 330 ohm resistor if your board is 5V)
 */
DFRobot_SHT20 sht20(&Wire, SHT20_I2C_ADDR);


void setup()
{
  Serial.begin(115200);

  // Init SHT20 Sensor
  sht20.initSHT20();
```

```
  delay(100);
  Serial.println("Sensor init finish!");

  /**
   * Check the current status information of SHT20
   * Status information: End of battery, Heater enabled, Disable OTP reload
   * Check result: yes, no
   */
  sht20.checkSHT20();
}

void loop()
{
  /**
   * Read the measured data of air humidity
   * Return the measured air humidity data of float type, unit: %
   */
  float humd = sht20.readHumidity();

  /**
   * Read the measured temp data
   * Return the measured temp data of float type, unit: C
   */
  float temp = sht20.readTemperature();

  Serial.print("Time:");
  Serial.print(millis());   // Get the system time from Arduino
  Serial.print(" Temperature:");
  Serial.print(temp, 1);   // Only print one decimal place
  Serial.print("C");
  Serial.print(" Humidity:");
  Serial.print(humd, 1);   // Only print one decimal place
```

```
  Serial.print("%");
  Serial.println();

  delay(1000);
}
```

4.2 Serial Monitor Output

```
Sensor init finish!
End of battery: yes
Heater enabled: yes
Disable OTP reload: yes
Time:36443 Temperature:998.0C Humidity:998.0%
Time:37446 Temperature:998.0C Humidity:998.0%
Time:38448 Temperature:998.0C Humidity:998.0%
Time:39449 Temperature:998.0C Humidity:998.0%
Time:48380 Temperature:998.0C Humidity:998.0%
Time:49382 Temperature:998.0C Humidity:998.0%
Time:50383 Temperature:998.0C Humidity:998.0%
Time:51385 Temperature:998.0C Humidity:998.0%
Time:52386 Temperature:998.0C Humidity:998.0%
Time:53388 Temperature:998.0C Humidity:998.0%
Time:54389 Temperature:998.0C Humidity:998.0%
Time:55391 Temperature:998.0C Humidity:998.0%
Time:56392 Temperature:998.0C Humidity:998.0%
Time:57394 Temperature:998.0C Humidity:998.0%
```

Figure 4.1: Serial Monitotr

Chapter 5

Result

5.1 Result

Upon completion of the project, the integrated system successfully measures and displays real-time humidity and temperature data with high accuracy and reliability. Utilizing the SHT20 sensor interfaced with Arduino, the system provides instant feedback on environmental conditions, enabling users to monitor and analyze changes in humidity and temperature levels.